

# 실습으로 배우는 AWS

00 수업소개





## 1. AWS 시작하기

AWS (Amazon Web Services)의 개념과 AWS의 주요 서비스를 알아보고, 클라우드 서비스를 사용하기 위한 기본 구성에 대해 학습합니다.

### 2. AWS 컴퓨팅 서비스 활용하기

AWS의 핵심 컴퓨팅 서비스인 EC2에 대해 알아보고, 고가용성과 탄력적인 서비스를 위한 아키텍처와 구성방식에 대해 알아봅니다.



## 3. AWS Storage, DB 서비스 활용하기

확장성과 신뢰성을 가진 S3 스토리지 서비스와, RDBMS 기반의 RDS를 통한 백엔드 구성에 대해 알아보고, EC2와 연계하는 방법에 대해 배워봅니다.

## 수 4. AWS 네트워킹 활용하기

AWS VPC, Route 53의 개념을 알아보고, 실제 웹 애플리케이션 서비스 제공을 위한 도메인, NAT, ACL 구성을 하는 방법을 배워봅니다.



## AWS를 처음 입문한 개발자, 엔지니어 지망생

AWS를 실무에 도입하고자 하는 개발자

AWS환경 서비스 구축을 위한 기획자



## AWS 주요 서비스에 대한 이해

AWS 서비스를 직접 구성할 수 있는 핸즈온 스킬

AWS 환경에서 웹 애플리케이션 구축 및 활용



# 실습으로 배우는 AWS

**01** AWS 시작하기



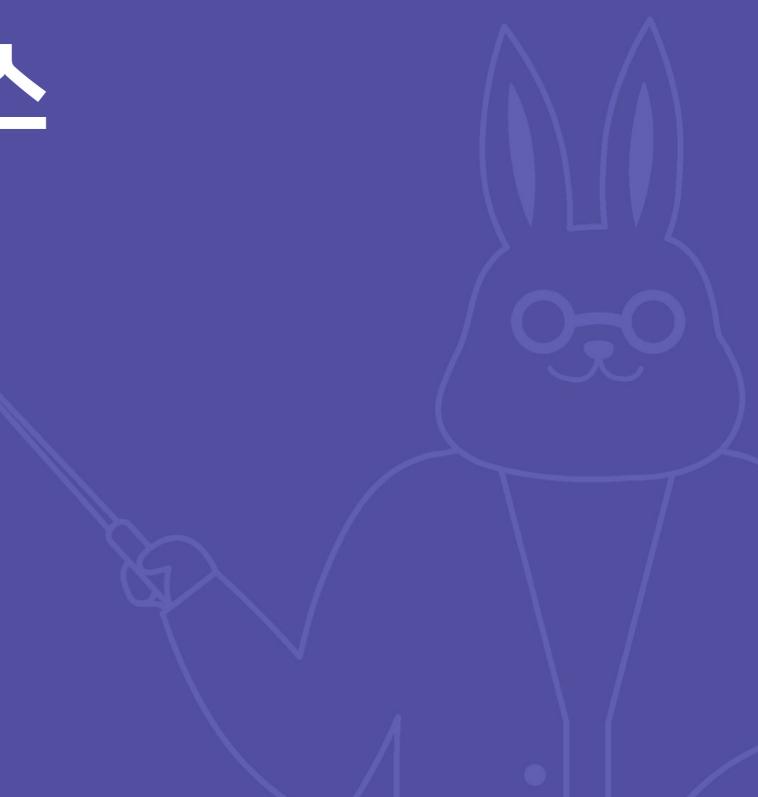
Copyright Elice. All Rights Reserved



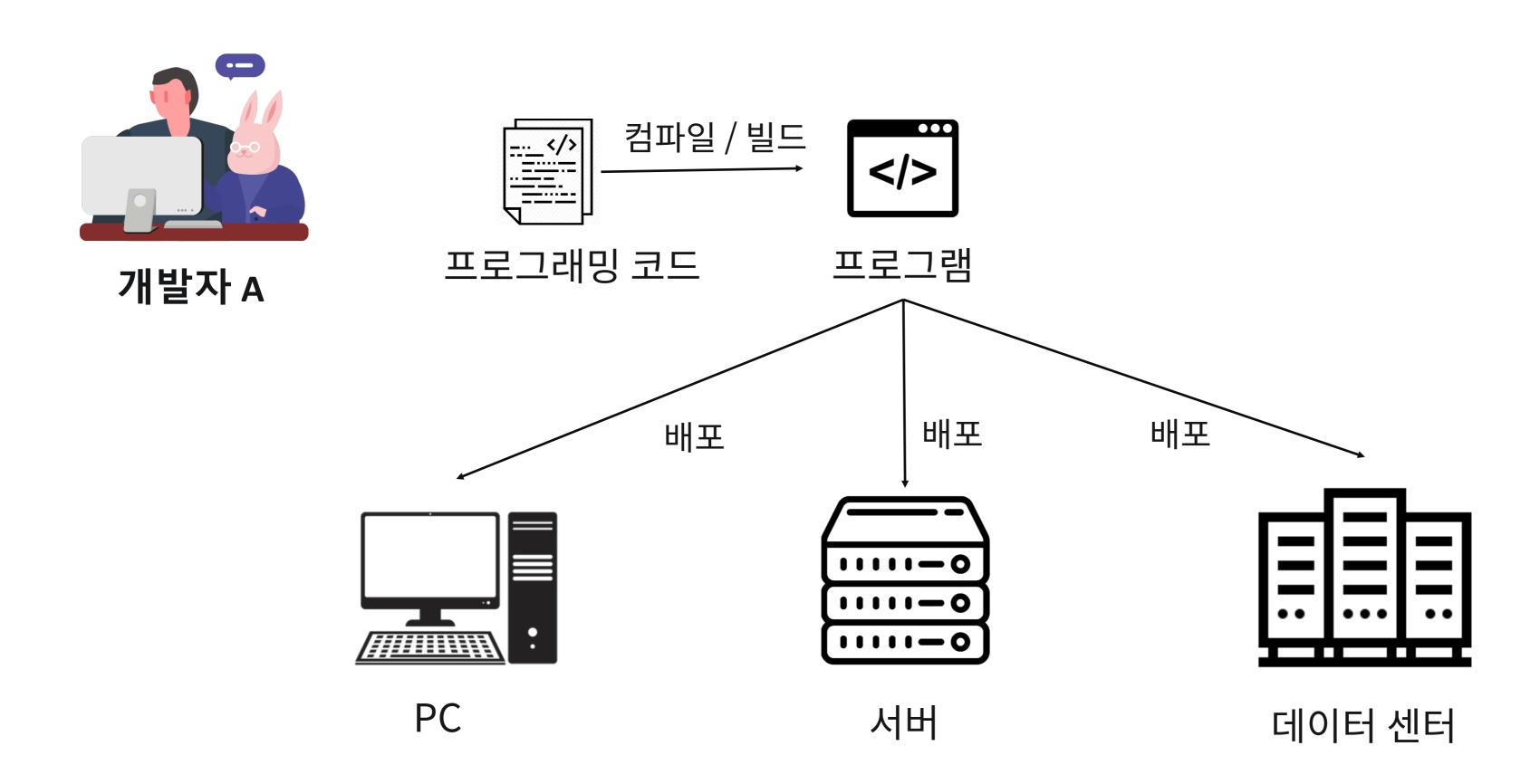
- 01. AWS 개념 & 주요 서비스
- 02. AWS 글로벌 인프라
- 03. AWS IAM 서비스 소개
- 04. AWS 서비스 사용법
- 05. [실습] AWS 계정 생성 및 IAM 구성하기

01

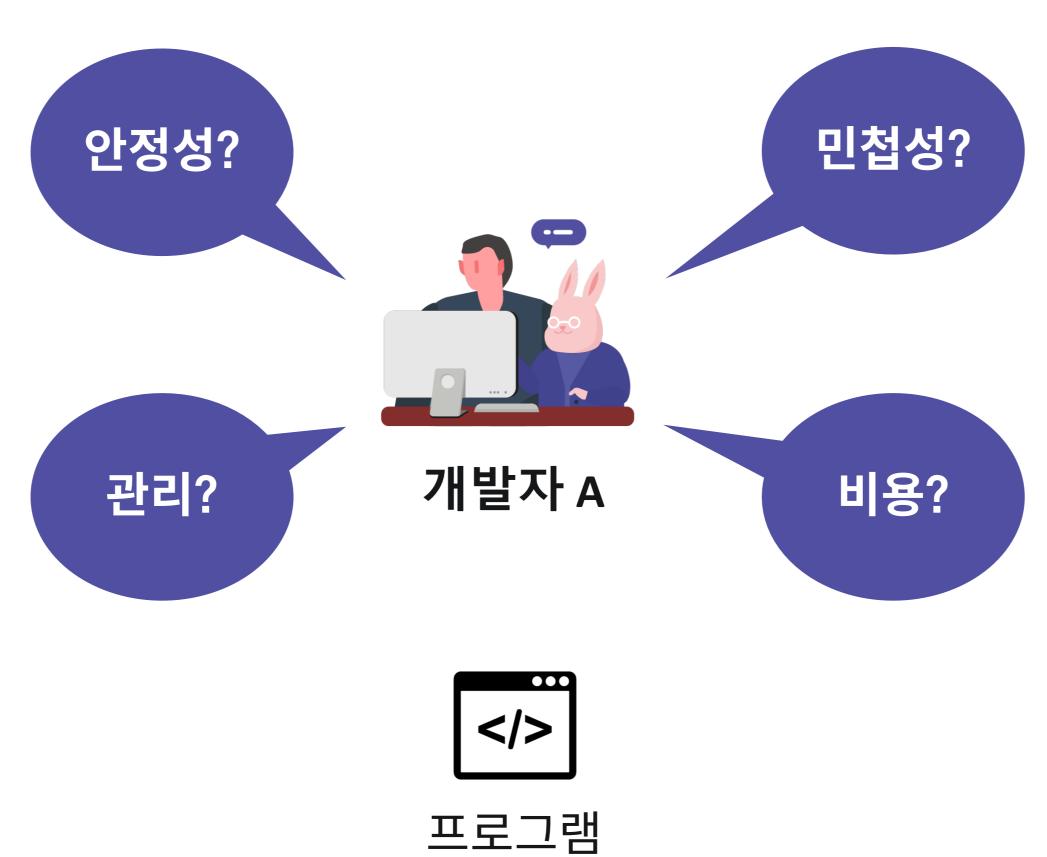
## AWS 개념 & 주요 서비스



#### ❷ 클라우드란 무엇일까? -1



### ❷ 클라우드란 무엇일까? -2

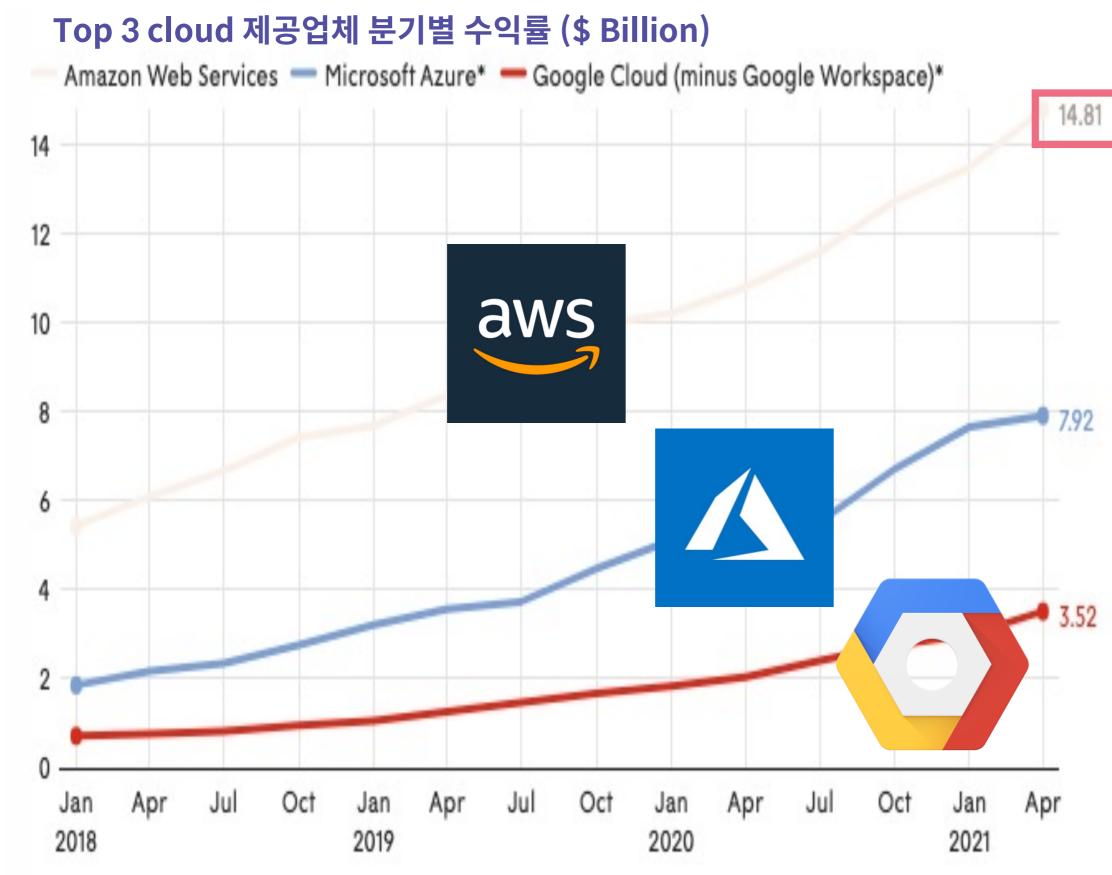


● 아마존 웹 서비스 (Amazon web services: AWS)



글로벌 #1 클라우드 서비스 제공 업체 300개 이상의 온 디맨드 (On-Demand) 클라우드 서비스 넷플릭스, 페이스북, 트위치 등 수 많은 기업에서 AWS 사용 중

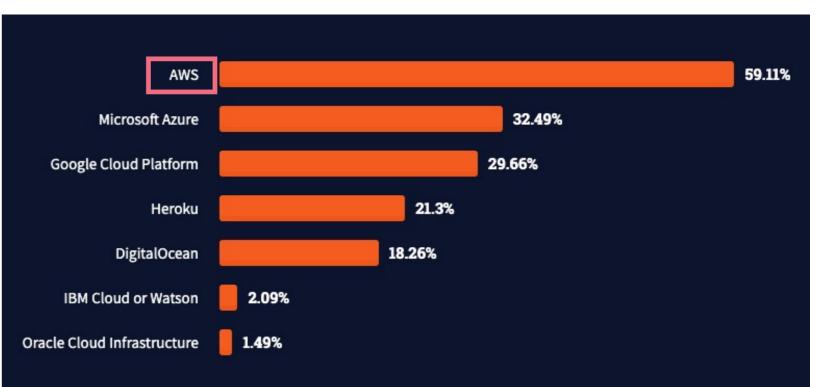
### ☑ 클라우드 서비스 제공업체 비교 (Amazon Web services, Microsoft Azure, Google cloud)



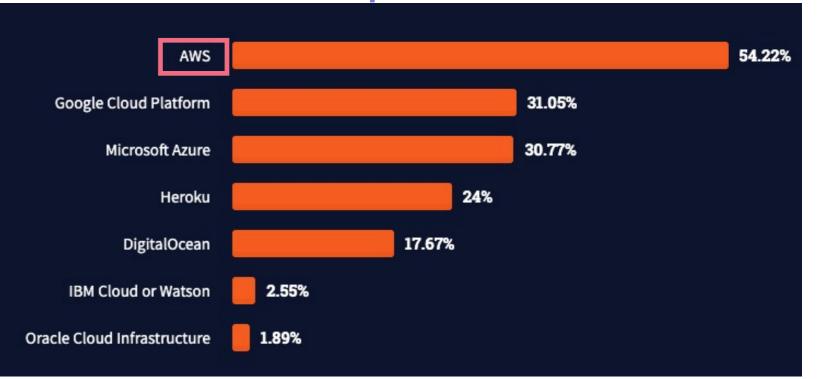
출처: Forrester Estimates: top 3 cloud revenues

https://techmonitor.ai/technology/cloud/aws-heading-50-public-cloud-market-share

#### **Professional Developers**



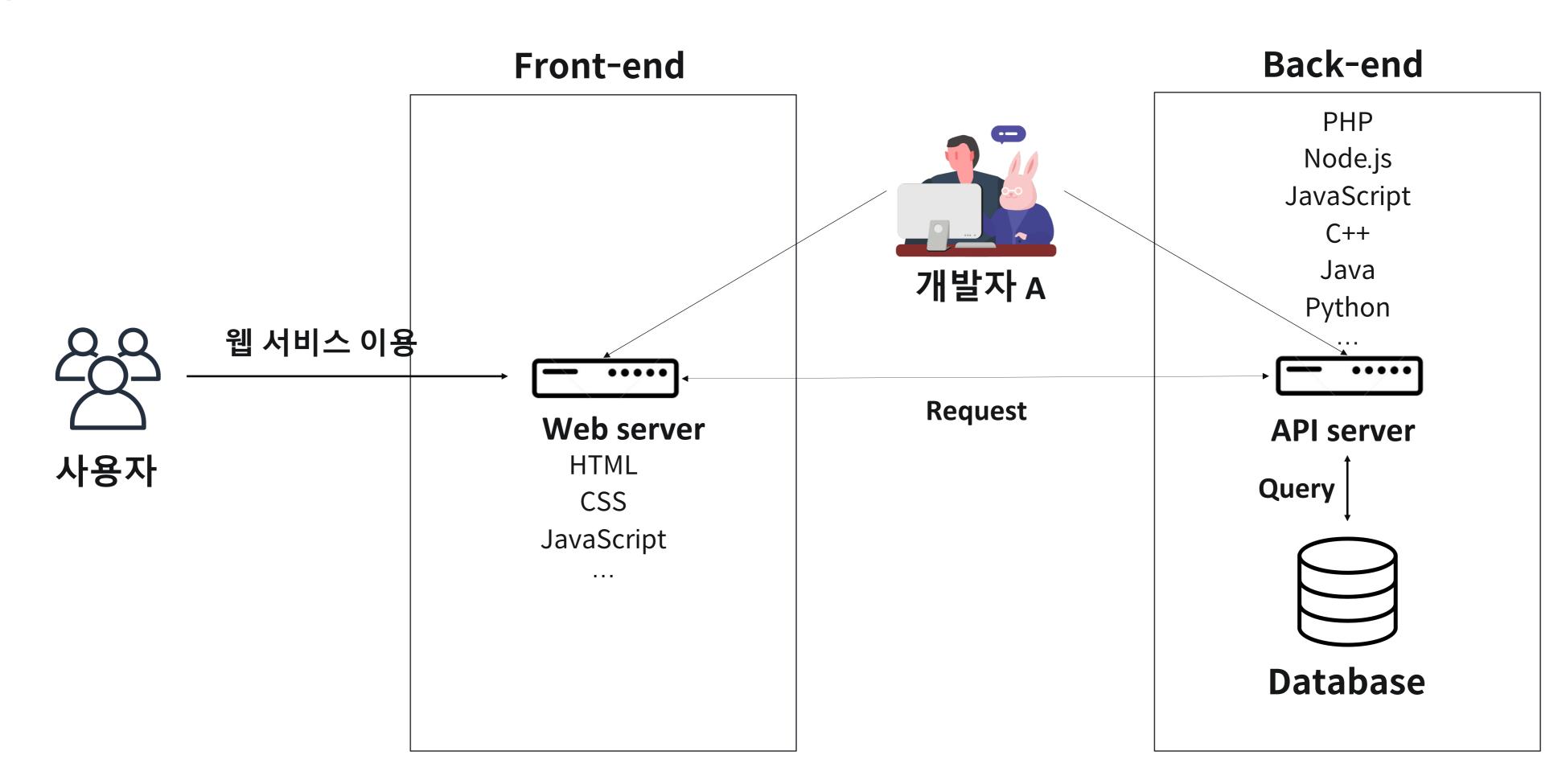
#### **All Respondents**



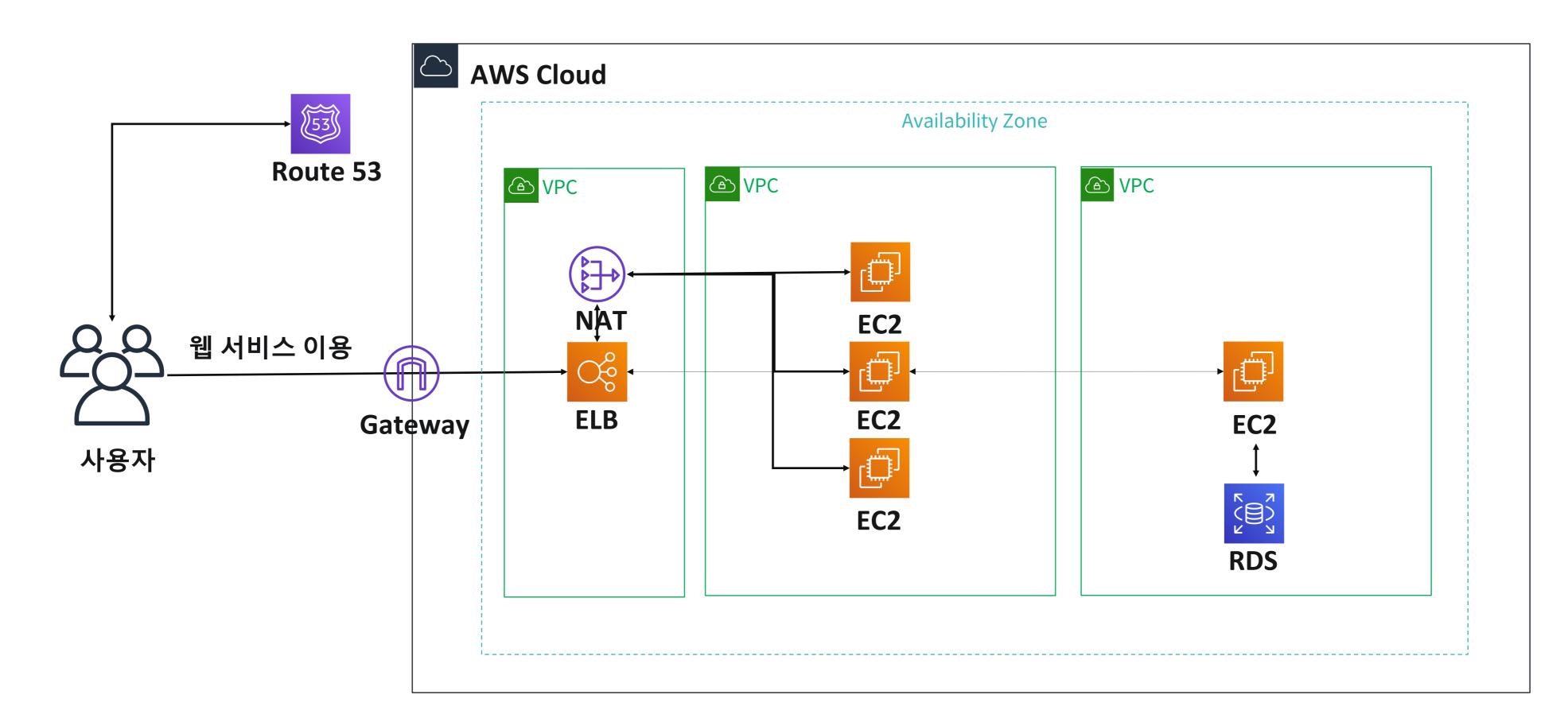
출처 : Stackoverflow

https://insights.stackoverflow.com/survey/2021#most-popular-technologies-platform

### ❷ 웹 어플리케이션 기본 구성(Front-end, Back-end)



### ❷ 웹 어플리케이션 기본 구성(Front-end, Back-end)



**01** AWS 개념 & 주요 서비스

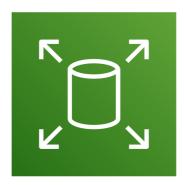
#### ❷ 이번 강의에서 배우게 될 주요 AWS 서비스



















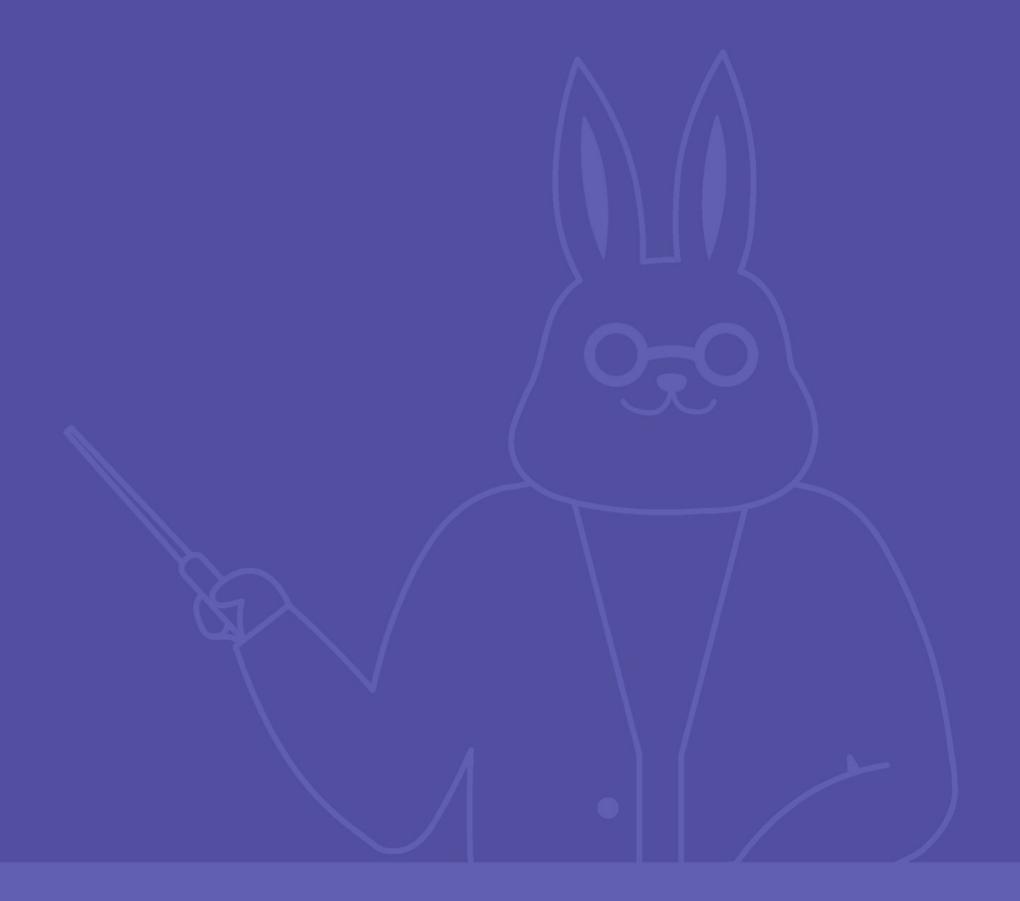




- 1. 클라우드는 빠르고, 안정적이면서, 비용 효율적인 컴퓨팅 서비스를 제공한다.
- 2. AWS는 클라우드 서비스 제공 업체로, 300개 이상의 온-디맨드 서비스를 제공한다.
- 3. 본 강의에서 사용되는 AWS 의 서비스들이 여러 가지가 존재한다.

02

# AWS 글로벌 인프라



#### ❷ AWS 글로벌 인프라스트럭처 – 245개 국가에서 서비스 운영



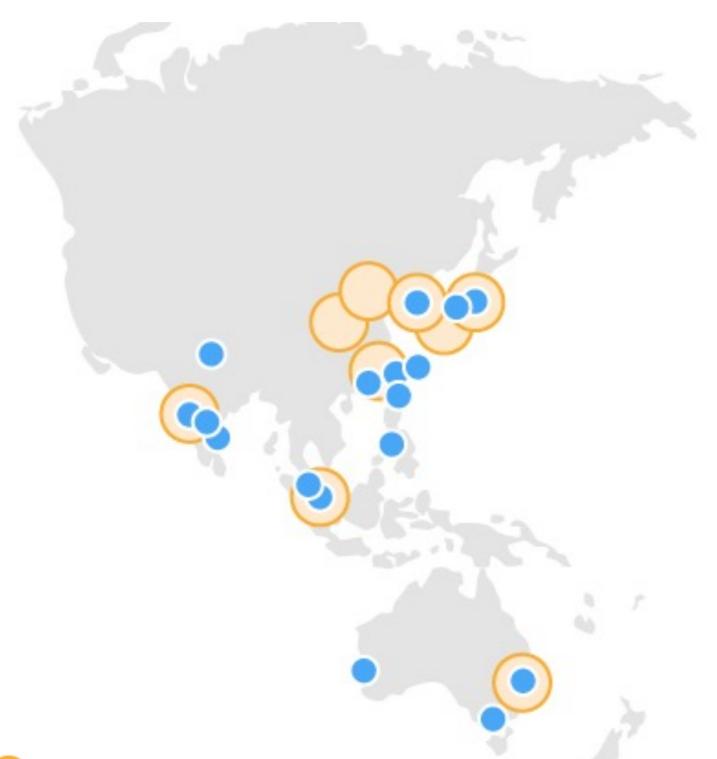
Region (리전)

Availability Zone (가용 영역)

Edge Location (엣지 로케이션)

출처 : https://infrastructure.aws

### ❷ AWS 리전 (Regions)



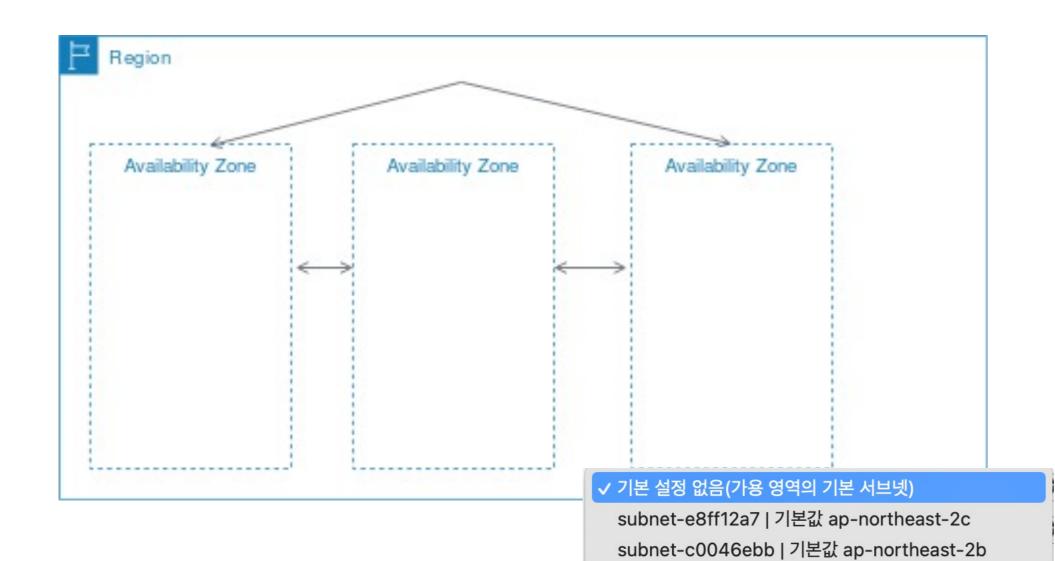
- O 리전 (Regions)
- 엣지 로케이션 (Edge Location)

- 리전:물리적인 데이터센터 클러스터 단위
- · 전 세계적으로 리전 단위로 AWS 서비스 제공
- 하나의 국가에 한 개 혹은 다수의 리전 존재
- 아시아 태평양 리전 : 중국 (베이징, 닝샤), 시드니, 오사카, 도쿄, 뭄바이, 서울, 홍콩

#### ❷ AWS 가용 지역 (Availability Zones) : Seoul Region 에 총 4개의 AZ 존재

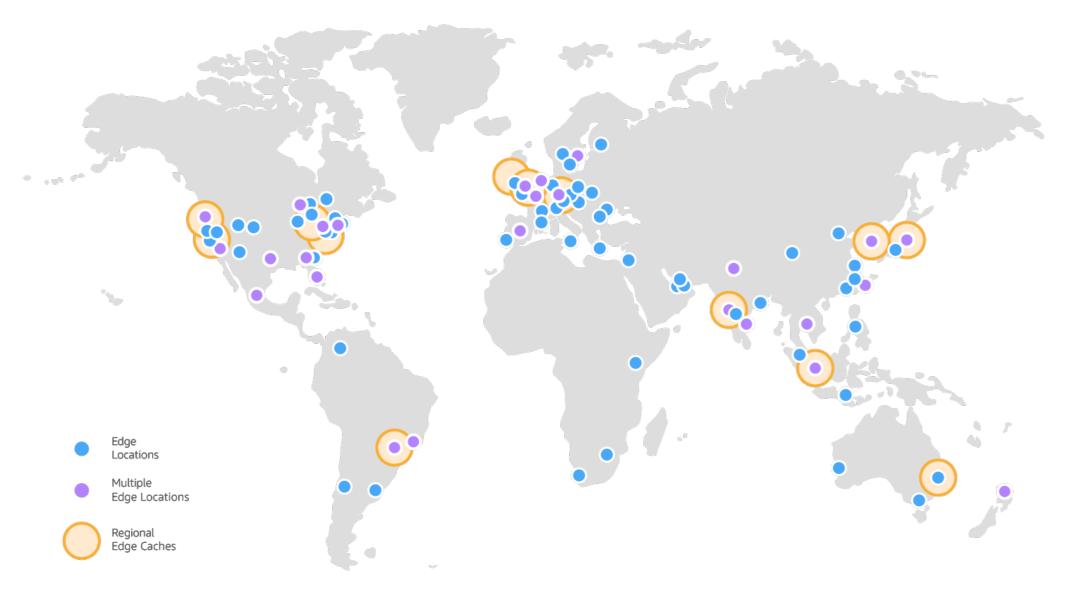
subnet-47ba4418 | 기본값 ap-northeast-2d

subnet-39bb0252 | 기본값 ap-northeast-2a



- 가용지역 (Availability Zone : AZ) :
   지리적으로 분리된 1개 이상의 <u>데이터센터</u>
- 서울 리전에는 총 4개의 AZ 존재
- ap-northeast-2a
- ap-northeast-2b
- ap-northeast-2c
- ap-northeast-2d
  - 왜 AZ 를 분리하였을까? : <u>내결함성과</u> 고가용성을 유지하기에 적합한 구조

### ♥ 엣지 로케이션 (Edge Location) : 더 나은 네트워크 속도를 위한 서비스



- 엣지 로케이션 (Edge Locations) : 전 세계적으로 통신사와 협업하여 빠른 네트워크 서비스를 제공하기 위한 지역
- 215개 이상의 엣지 로케이션을 **전 세계적으로** 제공
- 해외 서비스를 제공할 때 유용

☑ Region vs AZ vs Edge Location 이 왜 중요할까?





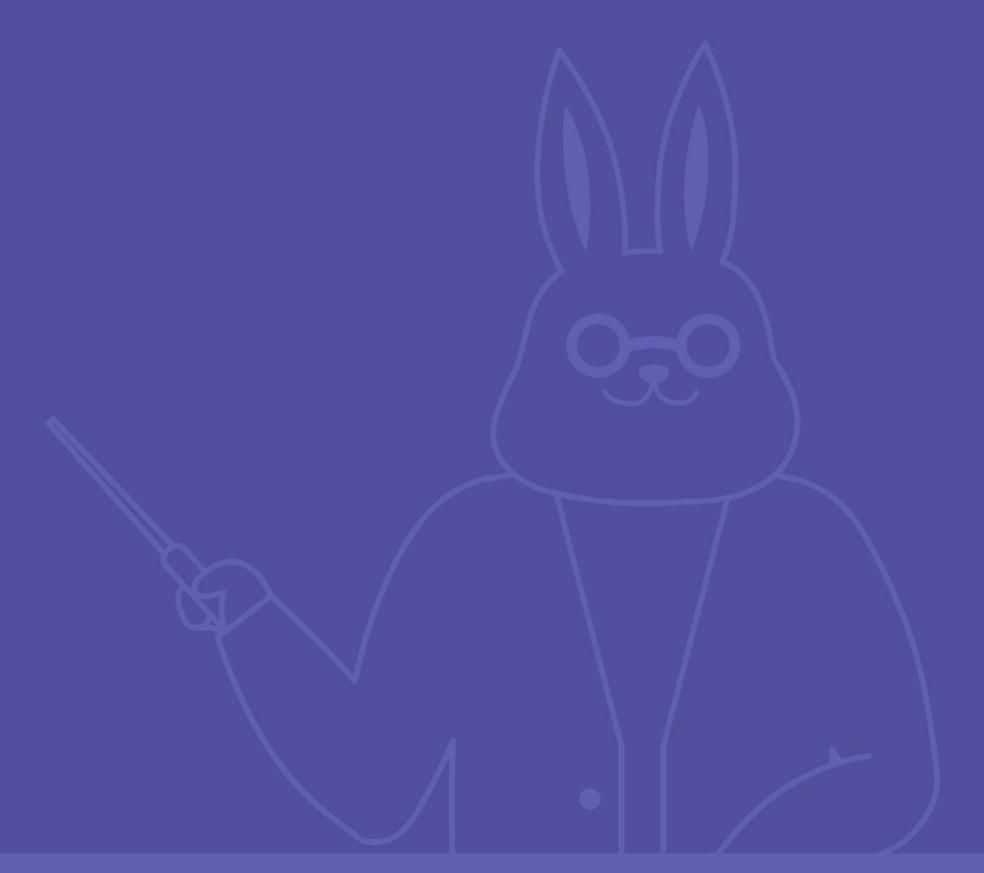




- 1. AWS 글로벌 인프라에는 Region, AZ, Edge Location 이라는 개념이 있다.
- 2. Region은 여러 개 데이터 센터의 묶음 단위, AZ는 실제 물리적인 데이터센터, Edge Location은 더 가까운 곳에서 빠른 네트워크 서비스를 위한 단위이다.
- 3. 각각의 지역은 필요한 서비스, 지연시간, 비용에 의해서 결정된다.

03

## AWS IAM 서비스 소개



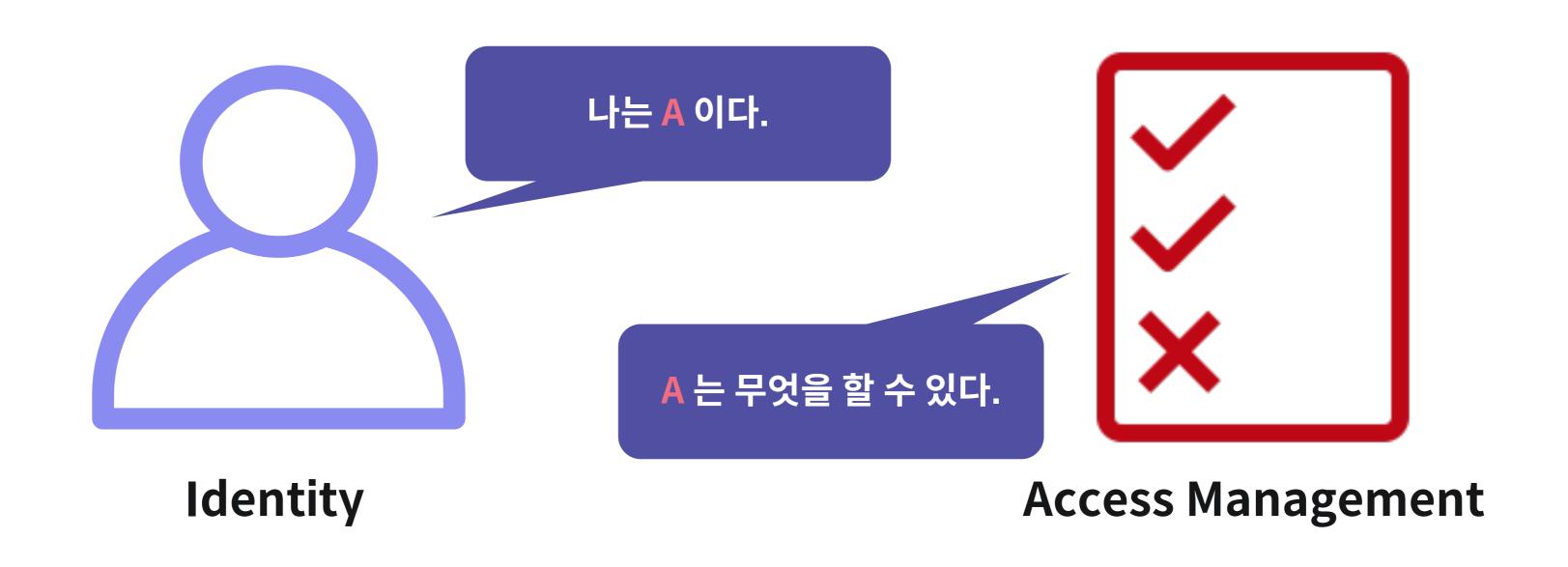
### ☑ IAM (Identity & Access Management): AWS의 자격 증명과 접근 관리 서비스!





AWS IAM (Identity and Access Management): Identity & Access Management





IAM은 AWS 서비스 전체의 권한 통제 시스템

### ❷ Root 계정, 사용자, 사용자 그룹 : Identity



나는 비상용 열쇠! Root 계정

- - AWS 계정 생성 시 최초로 만들어지는 계정
  - 모든 서비스에 대한 권한을 가지고 있음
  - MFA (Multi Factor Authentication)를 통해 보안 강화 필요



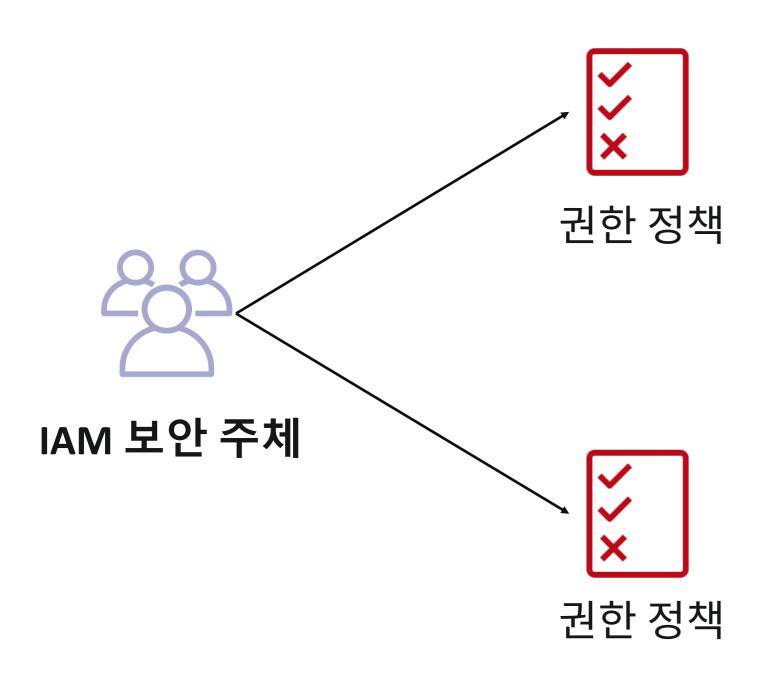
- AWS 서비스를 실질적으로 사용하는 유저, Root 계정과 독립적인 권한을 가짐
- 사용자는 필요한 최소한의 권한으로 가져야 함
- '그룹'에 의해서 그룹화 될 수 있음



- 여러 명의 사용자를 대상으로 그룹화를 하여 권한을 제공할 수 있음
- 사용자는 반드시 그룹에 소속될 필요는 없으며, 한 명의 사용자는 여러 개의 그룹에 소속될 수 있음

#### ☑ 권한 정책 (Permissions) : IAM Permission





- 정책 (Permission)은 사용자 혹은 사용자 그룹을 대상으로 적용할 수 있음
- 정책은 JSON 형식 기반으로 구성할 수 있음
- 하나의 사용자 혹은 사용자 그룹은 여러 개의 정책을 설정할 수 있음
- AWS 환경에서는 사용자에 대해 필요 이상의 권한을 부여하지 않는 것을 원칙으로 함

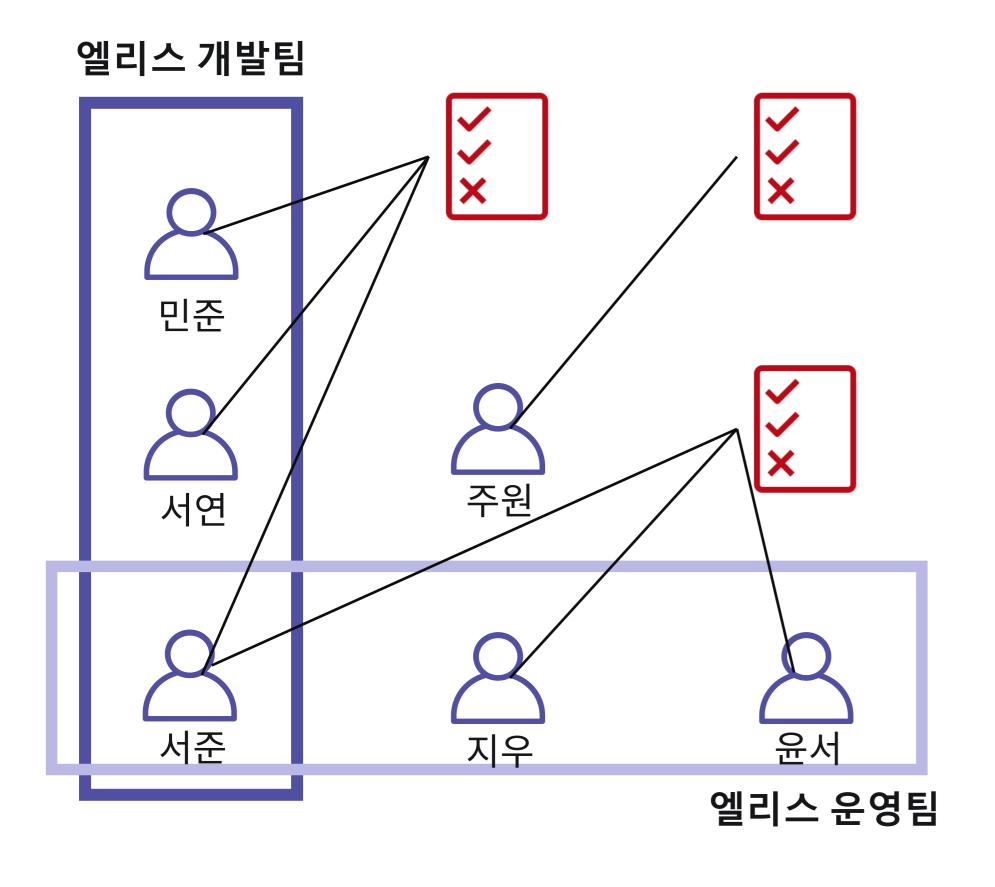
#### ☑ 정책 (Permission)의 JSON 형식: Permission JSON format



```
"Version": "2012-10-17"
                                                           Effect : 허용 & 차단?
"Statement": [
       "Effect": "Allow or Deny",
       "Action": [
                                                      Action : 어떤 행위를 할 것인가?
       "Resource": [
                                                   Resource : 어떤 객체 (리소스)들에 대해?
       "Condition": [
                                                        Condition : 어떤 조건에서?
```

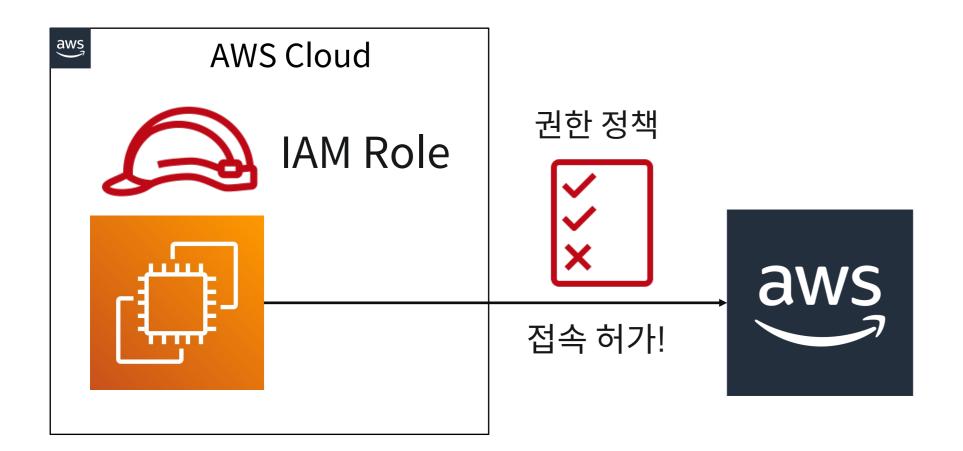
#### ☑ IAM: Identity (User, Group) + Access Management (Permission) 활용





- 1. 사용자 (User)는 여러 개의 그룹에 속할 수 있다. = 서준
- 2. 사용자 (User)는 반드시 그룹에 속하지 않아도 된다. = 주원
- 3. 사용자 (User)에게 여러 개의 정책이 적용될 수 있다. = 서준

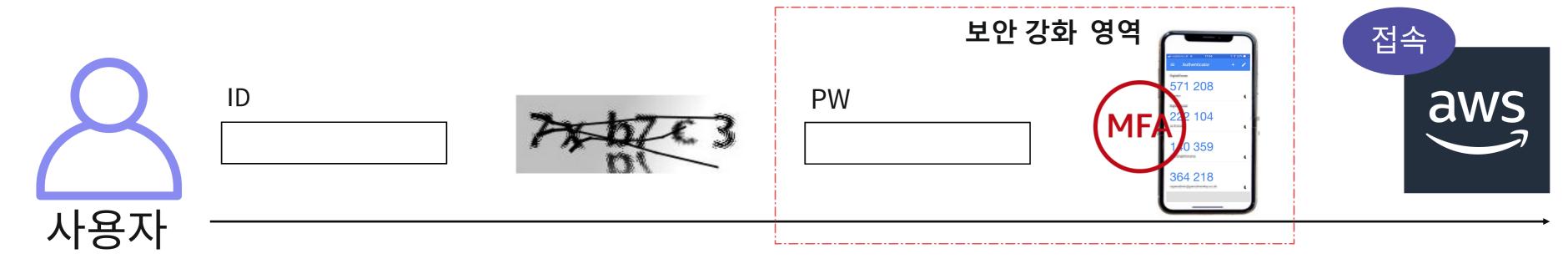




- 역할 (Role)은 AWS 서비스 대상으로 적용할 수 있음
- 역할 (Role)을 구성하기 위해서는 어떤 권한 (Permission)을 가지는지 명세 해야 함
- 역할 (Role)을 적용하는 주요 대상은 EC2, Lambda 등이 AWS 서비스를 호출하는 데 사용 됨(AWS 대부분의 서비스에 적용 가능)

#### ☑ 멀티 팩터 인증 (MFA: Multi Factor Authentication)

#### Login 과정



#### Password 정책

- 기본 암호 정책
- 최소 8자 이상 128자 암호 길이
- 대문자, 소문자, 특수문자 중에서 3개 이상의 조합
- AWS 계정 이름 혹은 이메일 주소와 동일하지 않아야 함
- \*\* 사용자가 지정하여 암호 정책을 변경 할 수 있음 (암호 최소 길이, 강도, 만료일, 재사용 권한 등)

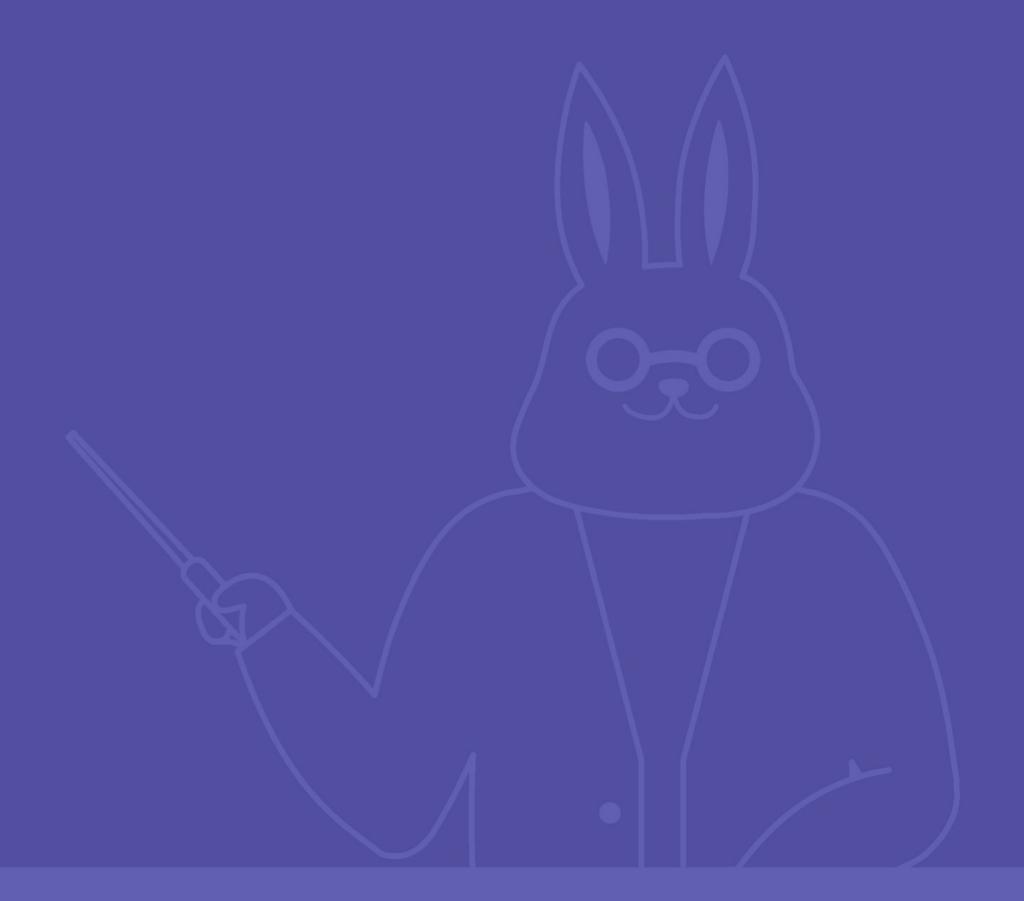
#### MFA 인증

- 비밀번호 접속만으로는 보안이 취약할 수 있기 때문에
   2중 암호 장치 (MFA)를 구성하는 방법이 제공됨
- MFA는 일정한 주기로 토큰이 갱신되며, 스마트폰 혹은 물리적인 기기를 통해 발급받을 수 있음
- Root User 나, IAM User 의 보안 강화를 할 수 있음

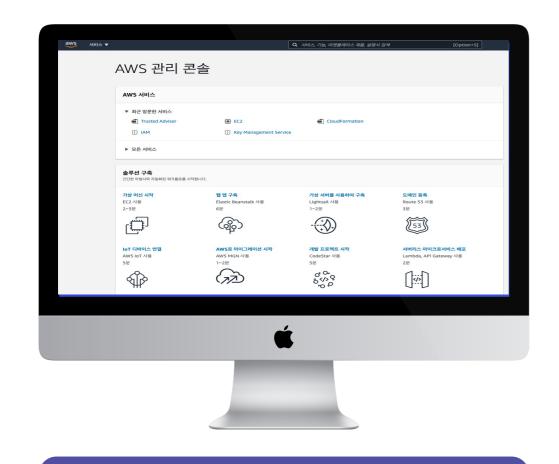
- 1. AWS IAM은 AWS 서비스 사용을 위한 Identity (신원)와 Access Management (권한 관리)를 위한 서비스 이다.
- 2. Identity에는 Root User, User, User Group 개념이 있다.
- 3. Permission은 User 나 User Group 대상으로 어떤 행위를 허용할 지를 명세한다.
- 4. Role은 AWS 서비스를 대상으로 적용할 수 있는 Permission이다.
- 5. 보안 강화를 위한 방법으로 암호 정책과 MFA가 있다.

04

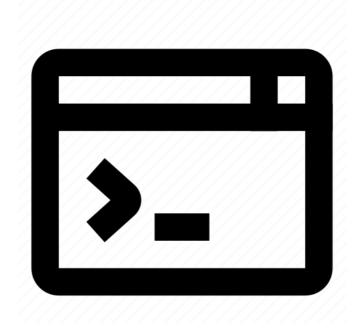
# AWS서비스사용법



### ☑ AWS 접속 방식



AWS 관리 콘솔 (Management Console)

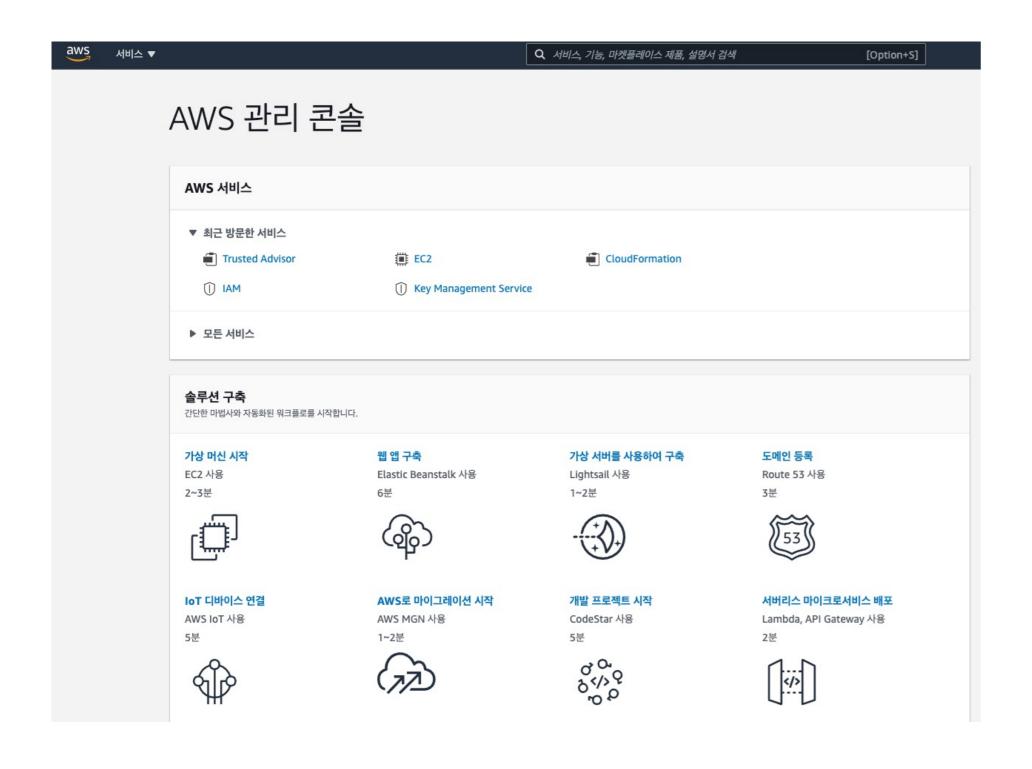


**AWS CLI** 



**AWS SDK** 

#### ✔ AWS 관리 콘솔 웹 인터페이스



AWS 관리 콘솔은 웹브라우저를 통해 접속하여 AWS 서비스를 제어할 수 있도록 제공하는 유저 인터페이스

사용자, 서비스 사용량, 월별 요금 등을 한 눈에 모니터링 가능

사용자에게 친숙하고, 자주 사용되는 서비스를 커스터마이징하여 관리 및 접근할 수 있음

#### ❷ AWS 액세스 키 & 비밀 액세스 키

▼ 액세스 키(액세스 키 ID 및 비밀 액세스 키)

Use access keys to make programmatic calls to AWS from the AWS CLI, Tools for PowerShell, AWS SDKs, or direct AWS API calls. You can have a maximum of two access keys (active or inactive) at a time.

보호를 위해 비밀 키를 다른 사람과 공유해서는 안 됩니다. 또한 키를 자주 교체하는 것이 좋습니다.

비밀 키는 생성 시에만 보거나 다운로드할 수 있습니다. 기존 비밀 키를 잘못 보관한 경우 새 액세스 키를 생성하십시오. 자세히 알아보기

생성 완료	액세스 키 ID	마지막 사용	마지막으로 사용한 리전	마지막으로 사용한 서비스	상태	작업
9월 7일 2021		해당 사항 없음	해당 사항 없음	해당 사항 없음	활성	비활성화   삭제

#### 새 액세스 키 만들기

루트 사용자 액세스 키는 전체 AWS 계정에 대한 무제한 액세스를 제공합니다. 장기 액세스 키가 필요한 경우 제한된 권한이 부여된 새 IAM 사용자를 생성하고 해당 사용자에 대한 액세스 키를 루트 사용자 액세스 키 대신 생성하는 것이 좋습니다. 자세히 알아보기

#### 액세스 키 & 비밀 액세스 키

- 액세스 키는 AWS CLI 나 AWS SDK 활용 시, ID 와 PW 같이 사용됨
- 액세스 키 ID = ID, 비밀 액세스 키 = Password
- 외부로 절대 노출 돼서는 안됨!



#### AWS CLI Shell 화면

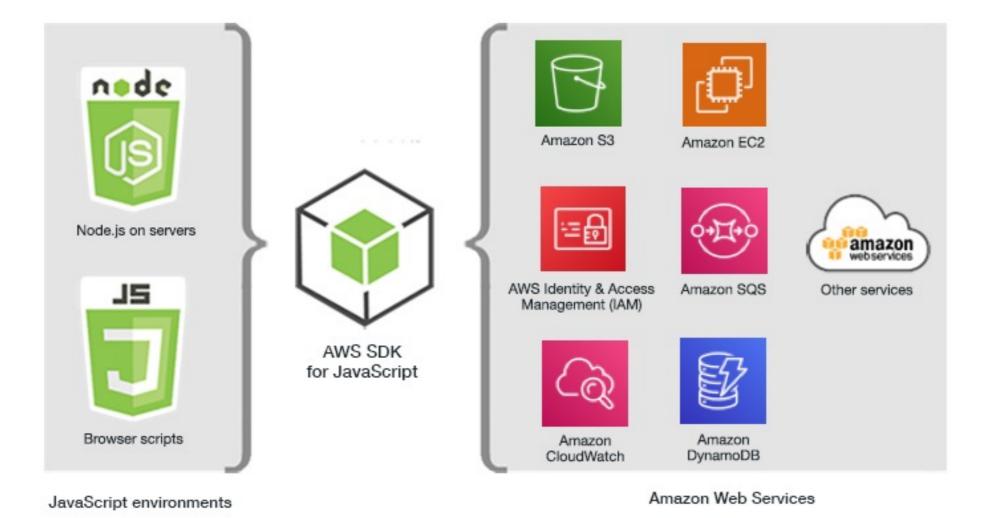


- AWS CLI는 커맨드 라인 방식으로 Shell 화면에서 AWS 서비스를 사용할 수 있도록 제공 해주는 툴
- AWS 관리 콘솔에서 수행하는 대부분의 역할을 수행 가능
- 커맨드 라인 스크립트 생성을 통해 반복적인 작업에 대해서는 <mark>스크립트 자동화</mark>를 구성할 수 있음
- AWS CLI 는 Open source 기반 : https://github.com/aws/aws-cli

04 AWS 서비스 사용법

#### AWS SDK

#### **AWS SDK**



- AWS SDK (Software Development Kit)는 프로그래밍 언어에서 라이브러리 형태로 제공되는 개발 킷
- AWS 서비스를 프로그래밍 언어를 통해 구현 및 사용할 수 있음
- 지원 언어 : C++, Go, Java, JavaScript, .NET, Node.js, PHP, Python, Ruby



- AWS 서비스를 사용하는 방식에는 크게 3가지가 있다.
   1) AWS 관리 콘솔, 2) AWS CLI, 3) AWS SDK
- 2. AWS CLI 와 AWS SDK 는 서비스를 사용 할 때 액세스 키와 비밀 액세스 키를 사용하므로 노출되지 않도록 주의한다.
- 3. AWS CLI는 커맨드 라인 방식으로 쉘에서 AWS 서비스를 사용할 수 있는 툴이다.
- 4. AWS SDK는 프로그래밍 언어를 통해 라이브러리와 같이 구현할 수 있는 개발 킷 이다.

05

## [실습] AWS 계정 생성 및 IAM 구성하기





AWS 계정 생성하기 AWS 글로벌 인프라 둘러보기 AWS 관리 콘솔 둘러보기

AWS IAM 구성하기 AWS CLI / SDK 구성하기

## 크레딧

/\* elice \*/

코스 매니저 임승연

콘텐츠 제작자 Jason

강사 Jason

감수자 장석준

디자이너 김루미

## 연락처

#### TEL

070-4633-2015

#### WEB

https://elice.io

#### E-MAIL

contact@elice.io

